

## **TAP Web Service**

As noted above, the TAP database was created expressly for the purpose of storing utility tariff data and calculating customer bills using data from these tariffs. The web service consists of several functions, described below. These functions provide a client with enough data to ultimately select and generate a monthly bill from any tariff in the TAP database. In order to integrate this capability with the current HES web site, a web service interface was added to the TAP database. This web service was implemented using the SOAP protocol.

Three groups of methods provide an interface with the TAP database. These include: Utility Listing methods to allow the user to select his or her utility, Tariff Listing methods to further select individual rate schedules, and Bill Calculation methods to determine utility bills based on load data.

- Utility Listing Methods – Accept state or ZIP code data, and return corresponding utility names and codes
- Tariff Listing Methods – Return a list of available tariffs for a particular utility
- Bill Calculation Methods – Return consumption, demand and fixed charges from consumption values generated by load processing module

A complete description of the SOAP schema and sample XML syntax can be found in Appendices F and G.

## **OnTAP SOAP Server Interface Description**

The OnTAP server accepts HTTP POST request in the form of well formed XML. These requested are then directed to a specific public function of the OnTAP server. Specific functions are declared in the request XML. Requests that are do not call a registered method will return the WSDL description of the OnTAP class.

The OnTAP class' public methods function as follows:

### **Utility Listing Methods**

The following methods accept different inputs and return the same utility information output as a XML document. This document is in the form of an array of string indexed arrays values (a 2 dimentional hash table).

- *doGetUtilityListByState*: This accepts either a state's full name or it's 2 letter abbreviation as a string.
- *doGetUtilityListByZip*: This accepts a 5 digit zip code as an integer.

The XML returned contains the following values:

- **name**: the name of the utility

- **util\_id**: the internal TAP utility id
- **state**: the state in which the utility is headquartered
- **country**: the country in which the utility operates
- **url**: a link to the utility's home page within TAP
- **eia\_code**: a unique code given to each utility in the United States by the E.I.A.

The key value returned for each utility is util\_id, which can then be used to run the tariff Listing Methods.

## **Tariff Listing Methods**

*doGetUtilityTariffs*: This accept one value: util\_id as a string. The XML returned contains the following values:

- **util\_id**: the internal TAP utility id
- **tariff\_id**: the internal TAP tariff id
- **schedule**: the name of the tariff as named by the utility
- **state**: the state for which the tariff is for
- **market**: the commercial market for which the tariff serves
- **service**: the service classification of the that the tariff delivers
- **commodity**: the commodity that the tariff covers
- **TOU**: is the tariff time-of -use or not
- **minDemand**: the minimum demand required by the customer to use the tariff as defined by the utility
- **maxDemand**: the maximum demand allowed by the customer as defined by the utility
- **url**: a link to the tariff's description page within TAP

The key value returned for each utility is tariff\_id, this is used to run the Bill Calculation Methods.

## **Tariff Description Methods:**

*doGetTOU*: Returns specific time-of-use information for the tariff \$tariff\_id. This accepts 1 value: tariff\_id as an integer. It returns the Time-of-use data for tariff\_id as follows: An array with items for each month of the year is returned. Each item contains the following.

- **monthName**: Is the name of a calendar month
- **peakDays**: Is an array of the days of the week for which the time-of-use period apply
- **hourEnding\_01 ... hourEnding\_24**: Is the time-of-use billing period, each of can have a value can of: onPeak, offPeak, or shoulder.

## **Bill Calculation Methods:**

*doGetMonthlyBill*: This accepts 8 values, all of them integers: tariff\_id, onPeakDemand, onPeakConsumption, shoulderDemand, shoulderConsumption, offPeakDemand, offPeakConsumption, and month. It returns bill information contained in a string indexed array. The bill data returned is the same for both time-of-use tariffs and standard block rate types. The values returned are as follows:

- **totalCharges**: the sum of all charges
- **consumptionTotal**: the sum of all consumption charges
- **demandTotal**: the sum of all demand charges
- **fixedTotal**: the sum of all fixed charges
- **offPeakConsumptionTotal**: the sum of all consumption charges that occur during the off-peak time-of-use period.
- **offPeakDemandTotal**: the sum of all demand charges that occur during the off-peak time-of-use period.
- **onPeakConsumptionTotal**: the sum of all consumption charges that occur during the peak time-of-use period.
- **onPeakDemandTotal**: the sum of all demand charges that occur during the peak time-of-use period.
- **shoulderConsumptionTotal**: the sum of all consumption charges that occur during the shoulder or partial-peak time-of-use period.
- **shoulderDemandTotal**: the sum of all demand charges that occur during the shoulder or partial-peak time-of-use period .
- **annualConsumptionTotal**: the sum of all other consumption charges that occur regardless of the time-of-use period.
- **annualDemandTotal**: the sum of all other demand charges that occur regardless of the time-of-use period.

*DoGetYearlyBill*: Is a wrapper for the doGetMonthlyBill method, it allows clients to send a complete year's worth of inputs and return a complete year's worth of bills. It accepts and returns a 12-element array (one for each month) of doGetMonthlyBill inputs and outputs.

## OnTAP Web Service Schema

Internally the Ontap server is written in PHP.

### Dependencies

- Apache >= 1.3 (<http://www.apache.org>)
- PHP >= 4.1.2 (<http://www.php.net>)
- PHP::PEAR::SOAP 0.8RC3 (<http://pear.php.net/soap>)

### Main Class

The OnTAP class is the object that is directly manipulated through the SOAP interface.

#### *OnTAP Class*

```
Init Ontap();
```

##### *Properties*

- Namespace: Is the SOAP server namespace
- UtiliyList: Is an array container for returning a list of utilities
- TariffListByUtility: Is an array container for returning a list of tariffs
- MonthlyBill: Is a container for returning monthly bill data
- YearlyBillData: Is an array container for returning a years worth of monthly bill data
- \_\_dispatch\_map: Is a private property used by the SOAP server. It is an array of the inputs and outputs by type; this is used for type checking and the generation of the WSDL file by the SOAP server.

##### *Methods*

- onTAP: Is the SOAP server constructor method
- DoGetMonthlyBill: Returns the values of the MonthlyBillResults class as a monthly bill.
- DoGetYearlyBill: Is the method called to generate a years worth of monthly bills by returning the values of the MonthlyBillResults class for each month.
- DoGetTOU: Returns the results from the TOUMonthData class
- DoGetUtilityListByState: Generates the input query and returns the values from UtilityList class for selecting utilities by state.
- DoGetUtilityListByZip: Generates the input query and returns the values from UtilityList class for selecting utilities by zip code.
- DoGetUtilityTariffs: Returns and array of Tariff class for each tariff of a utility
- \_\_dispatch\_map: Is a private method used by the SOAP server to create an instance of the \_\_dispatch\_map property. It is used for type checking inputs and outputs; it is also used to generate a WSDL file.

## OnTAP Sub-Classes

The following classes are manipulated via the OnTAP class.

#### *Utility Class*

```
Init Utility( string $name,
              int $util_id,
              int $eia_code,
              string $country,
```

```
    string $state );
```

*Properties*

- utility\_id: Is the TAP identification number for a utility.
- Name: Is the name of a utility
- eia\_code: Is the EIA's identification for a utility.
- country: Is the country a utility operates in.
- state: Is the U.S. state which a utility is head-quartered in.
- url: Is the url to the TAP description of the a utility.

*Methods*

- utility: Is the constructor method
- \_fixEIA: is a private method which translates the TAP encoded eia\_code into the public eia\_code property
- \_makeURL: Is a private method which sets the URL property.

### *TOU Data Class*

```
Init TOUData( int $tariff_id )
```

*Properties*

- TOUMonths: is an array of TOU Month Data classes.

*Methods*

- TOUData: Is the consturctor mehtod.

### *TOU Month Data Class*

```
Init TOUMonthData( int $tariff_id, int $month );
```

*Properties*

- MonthName: Is the name of a calendar month
- Season: Is the name of the season that the MonthName property is in, as defined by the tariff \$tariff\_id.
- PeakDays: Is an array of the days of the week for which the time-of-use period apply
- HourEnding\_01 ... HourEnding\_24: Is the time-of-use billing period, each of can have a value can of: onPeak, offPeak, or shoulder.
- \_TOUHours: Is a private property whchich contains a copy of the TOU Hours class.
- \_tariff\_id: Is a private copy of tariff\_id.
- \_utility\_id: Is a private copy of utility\_id.

*Methods*

- TOUMonthData: Is the constuctor method
- \_SetTariffID: Is a private method used to check and set the private \_tariff\_id property
- \_SetUtilityID: Is a private method used to check and set the private \_utility\_id property
- \_SetMonthData: Is a private method that sets the MonthName property.
- SetSeason: Is a private method used which returns the season as defined by the tariff \$tariff\_id for the value of the MonthName property.

### *TOU Hours Class*

```
Init TOUHours( int $tariff_id, string $season );
```

*Properties*

- PeakDays: Is an array of the days of the week for which the time-of-use period apply
  - HourEnding\_01 ... HourEnding\_24: Is the time-of-use billing period, each of can have a value can of: onPeak, offPeak, or shoulder.

*Methods*

- TOUHours: Is the constructor class
- \_seasonAbbreviationToWords: Is a private method that converts the internal TAP storage of season names into a human readable string.
- \_peak\_daysToDays: Is a private method that populates the PeakDays property

*Utility List Class*

```
Init UtilityList( string $sql );
```

*Properties*

- utilityList: Is an array of Utility classes
- size: Is the length of the utilityList array
- error: Is a Boolean value of whether or not an error occurred during the utilityList instance
- errorMessage: Is the error message of any error that occurred during the utilityList instance.

*Methods*

- utilityList: Is the constructor method, it takes a SQL query as its only variable
- utilityQuery: Queries the TAP database and returns the values to utilityList
- size: Is the number of elements in the utilityList array
- \_noResultsError: Is a private method that sets the error and errorMessage properties if no results are returned from utilityQuery.
- \_noConnectionError: Is a private method that sets the error and errorMessage properties if the class is unable to connect to the TAP database.

*Tariff Class*

```
Init Tariff( int $tariff_id,
             string $schedule,
             int $util_id,
             int $TOU,
             string $market,
             string $service,
             int $minDemand,
             int $maxDemand );
```

*Properties*

- tariff\_id: Is the TAP identification number for a tariff.
- schedule: Is the name of the tariff, as named by its parent utility.
- util\_id: Is the TAP identification number for a tariff's parent utility.
- TOU: Is a Boolean value of whether a tariff is a time-of-use type tariff or not.
- state: Is the U.S. state which a tariff is for.

- market: Is the market segment which a tariff serves.
- service: Is the type service which is delivered by a tariff.
- minDemand: Is the minimum peak demand a customer must have to qualify for this tariff.
- maxDemand: Is the maximum peak demand a customer must have to qualify for this tariff.
- url: Is the url to the TAP description of the a tariff.

#### *Methods*

- Tariff: Is the constructor method.
- isValidTariff: Determines whether the instantiated tariff exists in the TAP database or not.
- \_makeURL: Is a private method which sets the url property.

#### *Monthly Bill Results Class*

```
Init MonthlyBillResults(    int $tariff_id,
                            int $onPeakDemand,
                            int $onPeakConsumption,
                            int $shoulderDemand,
                            int $shoulderConsumption,
                            int $onPeakDemand,
                            int $offPeakDemand,
                            int $month );
```

#### *Properties*

- tariff\_id: Is the TAP identification number for a tariff.
- totalCharges: the sum of all charges
- consumptionTotal: the sum of all consumption charges
- demandTotal: the sum of all demand charges
- fixedTotal: the sum of all fixed chages
- offPeakConsumptionTotal: the sum of all consumption charges that occur during the off-peak time-of-use period.
- offPeakDemandTotal: the sum of all demand charges that occur during the off-peak time-of-use period.
- onPeakConsumptionTotal: the sum of all consumption charges that occur during the peak time-of-use period.
- onPeakDemandTotal: the sum of all demand charges that occur during the peak time-of-use period.
- shoulderConsumptionTotal: the sum of all consumption charges that occur during the shoulder or partial-peak time-of-use period.
- shoulderDemandTotal: the sum of all demand charges that occur during the shoulder or partial-peak time-of-use period .
- annualConsumptionTotal: the sum of all other consumption charges that occur regardless of the time-of-use period.
- annualDemandTotal: the sum of all other demand charges that occur regardless of the time-of-use period.
- month: the month for which the returned charges apply

#### *Methods*

MonthlyBillResults: Is the constructor method. When the class is constructed, the bill is calculated.

## Sample XML Syntax for Web Service Linking TAP and HES

### DoGetUtilityByState

#### *Request*

```
<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:ns4="urn:onTAP"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body>
    <ns4:doGetUtilityListByState>
      <state xsi:type="xsd:string">CA</state>
    </ns4:doGetUtilityListByState>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

#### *Response*

```
<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:ns4="http://soapinterop.org/xsd"
  xmlns:ns5="urn:onTAP"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body>
    <ns5:doGetUtilityListByStateResponse>
      <return xsi:type="ns4:SOAPStruct">
        <item>
          <name xsi:type="xsd:string">Alameda City of</name>
          <util_id xsi:type="xsd:string">40</util_id>
          <eia_code xsi:type="xsd:string">207</eia_code>
          <country xsi:type="xsd:string">UNITED STATES</country>
          <state xsi:type="xsd:string">California</state>
          <url
            xsi:type="xsd:string">http://tariffs.lbl.gov/utility/utility.php?util_id=40</url>
        </item>
        <item>
          <name xsi:type="xsd:string">Anaheim City of</name>
          <util_id xsi:type="xsd:string">93</util_id>
          <eia_code xsi:type="xsd:string">590</eia_code>
          <country xsi:type="xsd:string">UNITED STATES</country>
          <state xsi:type="xsd:string">California</state>
          <url
            xsi:type="xsd:string">http://tariffs.lbl.gov/utility/utility.php?util_id=93</url>
        </item>
        <item>
          <name xsi:type="xsd:string">Turlock Irrigation District</name>
          <util_id xsi:type="xsd:string">2974</util_id>
          <eia_code xsi:type="xsd:string">19281</eia_code>
          <country xsi:type="xsd:string">UNITED STATES</country>
          <state xsi:type="xsd:string">California</state>
          <url
            xsi:type="xsd:string">http://tariffs.lbl.gov/utility/utility.php?util_id=2974</url>
```

```

</item>
</return>
</ns5:doGetUtilityListByStateResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

doGetUtilityByZip
Request
<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:ns4="urn:onTAP" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

<SOAP-ENV:Body>
<ns4:doGetUtilityListByZip>
<zipcode xsi:type="xsd:string">95841</zipcode>
</ns4:doGetUtilityListByZip>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

Response
<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:ns4="http://soapinterop.org/xsd" xmlns:ns5="urn:onTAP" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

<SOAP-ENV:Body>
<ns5:doGetUtilityListByZipResponse>
<return xsi:type="ns4:SOAPStruct">
<item>
<name xsi:type="xsd:string">Pacific Gas & Electric Co</name>
<util_id xsi:type="xsd:string">2207</util_id>
<eia_code xsi:type="xsd:string">14328</eia_code>
<country xsi:type="xsd:string">UNITED STATES</country>
<state xsi:type="xsd:string">California</state>
<url
xsi:type="xsd:string">http://tariffs.lbl.gov/utility/utility.php?util_id=2207</url>
</item>
<item>
<name xsi:type="xsd:string">Sacramento Municipal Util Dist</name>
<util_id xsi:type="xsd:string">2536</util_id>
<eia_code xsi:type="xsd:string">16534</eia_code>
<country xsi:type="xsd:string">UNITED STATES</country>
<state xsi:type="xsd:string">California</state>
<url
xsi:type="xsd:string">http://tariffs.lbl.gov/utility/utility.php?util_id=2536</url>
</item>
</return>
</ns5:doGetUtilityListByZipResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

## **doGetUtilityTariffs**

### *Request*

```
<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:ns4="urn:onTAP"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body>
    <ns4:doGetUtilityTariffs>
      <util_id xsi:type="xsd:string">1656</util_id>
    </ns4:doGetUtilityTariffs>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

### *Response*

```
<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:ns4="urn:onTAP"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body>
    <ns4:doGetUtilityTariffsResponse>
      <return xsi:type="SOAP-ENC:Array" SOAP-ENC:arrayType="Struct[2]" SOAP-
        ENC:offset="[0]">
        <item>
          <tariff_id xsi:type="xsd:string">1074</tariff_id>
          <schedule xsi:type="xsd:string">R-1 Rate B Time-of-Use Residential
            Service</schedule>
          <util_id xsi:type="xsd:string">1656</util_id>
          <TOU xsi:type="xsd:int">1</TOU>
          <state xsi:type="xsd:string">California</state>
          <market xsi:type="xsd:string">Residential</market>
          <service xsi:type="xsd:string">Residential</service>
          <minDemand xsi:type="xsd:string">0.0</minDemand>
          <maxDemand xsi:type="xsd:string">1000000000.0</maxDemand>
          <url
            xsi:type="xsd:string">http://tariffs.lbl.gov/tariff/tariff.php?util_id=1656&amp;
            ;tariff_id=1074</url>
          <commodity xsi:type="xsd:string">Electricity</commodity>
        </item>

        <item>
          <tariff_id xsi:type="xsd:string">1067</tariff_id>
          <schedule xsi:type="xsd:string">R-1 Rate A Residential Service</schedule>
          <util_id xsi:type="xsd:string">1656</util_id>
          <TOU xsi:type="xsd:int">0</TOU>
          <state xsi:type="xsd:string">California</state>
          <market xsi:type="xsd:string">Residential</market>
          <service xsi:type="xsd:string">Residential</service>
          <minDemand xsi:type="xsd:string">0.0</minDemand>
          <maxDemand xsi:type="xsd:string">1000000000.0</maxDemand>
          <url
            xsi:type="xsd:string">http://tariffs.lbl.gov/tariff/tariff.php?util_id=1656&amp;
            ;tariff_id=1067</url>
        </item>
      </return>
    </ns4:doGetUtilityTariffsResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```

<commodity xsi:type="xsd:string">Electricity</commodity>
</item>
</return>
</ns4:doGetUtilityTariffsResponse>

</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

## **doGetMonthlyBill**

### *Request*

```

<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" 
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:ns4="urn:onTAP"
    SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

<SOAP-ENV:Body>
<ns4:doGetMonthlyBill>
<tariff_id xsi:type="xsd:int">592</tariff_id>
<onPeakDemand xsi:type="xsd:int">198</onPeakDemand>
<onPeakConsumption xsi:type="xsd:int">897</onPeakConsumption>
<shoulderDemand xsi:type="xsd:int">456</shoulderDemand>
<shoulderConsumption xsi:type="xsd:int">245</shoulderConsumption>
<offPeakDemand xsi:type="xsd:int">325</offPeakDemand>
<offPeakConsumption xsi:type="xsd:int">645</offPeakConsumption>
<month xsi:type="xsd:int">7</month></ns4:doGetMonthlyBill>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

### *Response*

```

<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" 
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:ns4="http://soapinterop.org/xsd"
    xmlns:ns5="urn:onTAP"
    SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

<SOAP-ENV:Body>
<ns5:doGetMonthlyBillResponse>
<return xsi:type="ns4:SOAPStruct">
<tariff_id xsi:type="xsd:int">592</tariff_id>
<fixedTotal xsi:type="xsd:float">175</fixedTotal>
<consumptionTotal xsi:type="xsd:float">229.11066</consumptionTotal>
<demandTotal xsi:type="xsd:float">4707.6</demandTotal>
<onPeakDemandTotal xsi:type="xsd:float">2336.4</onPeakDemandTotal>
<onPeakConsumptionTotal xsi:type="xsd:float">147.76281</onPeakConsumptionTotal>
<shoulderDemandTotal xsi:type="xsd:float">1208.4</shoulderDemandTotal>
<shoulderConsumptionTotal
xsi:type="xsd:float">24.49755</shoulderConsumptionTotal>
<offPeakDemandTotal xsi:type="xsd:float">0</offPeakDemandTotal>
<offPeakConsumptionTotal xsi:type="xsd:float">56.8503</offPeakConsumptionTotal>
<annualDemandTotal xsi:type="xsd:float">1162.8</annualDemandTotal>
<annualConsumptionTotal xsi:type="xsd:float">0</annualConsumptionTotal>
<totalCharges xsi:type="xsd:float">5111.71066</totalCharges>
<month xsi:type="xsd:int">7</month></return></ns5:doGetMonthlyBillResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

## **doGetYearlyBill**

### *Request*

```
<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:ns4="urn:onTAP"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body>
    <ns4:doGetYearlyBill>
      <YearlyBillInputs xsi:type="SOAP-ENC:Array" SOAP-ENC:arrayType="Struct[12]"
        SOAP-ENC:offset="[0]">
        <item>
          <tariff_id xsi:type="xsd:int">299</tariff_id>
          <onPeakDemand xsi:type="xsd:int">198</onPeakDemand>
          <onPeakConsumption xsi:type="xsd:int">897</onPeakConsumption>
          <shoulderDemand xsi:type="xsd:int">456</shoulderDemand>
          <shoulderConsumption xsi:type="xsd:int">245</shoulderConsumption>
          <offPeakDemand xsi:type="xsd:int">325</offPeakDemand>
          <offPeakConsumption xsi:type="xsd:int">645</offPeakConsumption>
          <month xsi:type="xsd:int">1</month>
        </item>

        <item>
          <tariff_id xsi:type="xsd:int">299</tariff_id>
          <onPeakDemand xsi:type="xsd:int">198</onPeakDemand>
          <onPeakConsumption xsi:type="xsd:int">897</onPeakConsumption>
          <shoulderDemand xsi:type="xsd:int">456</shoulderDemand>
          <shoulderConsumption xsi:type="xsd:int">245</shoulderConsumption>
          <offPeakDemand xsi:type="xsd:int">325</offPeakDemand>
          <offPeakConsumption xsi:type="xsd:int">645</offPeakConsumption>
          <month xsi:type="xsd:int">2</month>
        </item>

        ... For each month ...

        <item>
          <tariff_id xsi:type="xsd:int">299</tariff_id>
          <onPeakDemand xsi:type="xsd:int">198</onPeakDemand>
          <onPeakConsumption xsi:type="xsd:int">897</onPeakConsumption>
          <shoulderDemand xsi:type="xsd:int">456</shoulderDemand>
          <shoulderConsumption xsi:type="xsd:int">245</shoulderConsumption>
          <offPeakDemand xsi:type="xsd:int">325</offPeakDemand>
          <offPeakConsumption xsi:type="xsd:int">645</offPeakConsumption>
          <month xsi:type="xsd:int">11</month>
        </item>

        <item>
          <tariff_id xsi:type="xsd:int">299</tariff_id>
          <onPeakDemand xsi:type="xsd:int">198</onPeakDemand>
          <onPeakConsumption xsi:type="xsd:int">897</onPeakConsumption>
          <shoulderDemand xsi:type="xsd:int">456</shoulderDemand>
          <shoulderConsumption xsi:type="xsd:int">245</shoulderConsumption>
          <offPeakDemand xsi:type="xsd:int">325</offPeakDemand>
          <offPeakConsumption xsi:type="xsd:int">645</offPeakConsumption>
          <month xsi:type="xsd:int">12</month>
        </item>
      </YearlyBillInputs>
```

```

</ns4:doGetYearlyBill>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

*Response*

```

<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:ns4="http://soapinterop.org/xsd"
  xmlns:ns5="urn:onTAP"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body>
    <ns5:doGetYearlyBillResponse>
      <return xsi:type="ns4:SOAPStruct">
        <item>
          <tariff_id xsi:type="xsd:int">299</tariff_id>
          <fixedTotal xsi:type="xsd:float">0</fixedTotal>
          <consumptionTotal xsi:type="xsd:float">232.33067</consumptionTotal>
          <demandTotal xsi:type="xsd:float">0</demandTotal>
          <onPeakDemandTotal xsi:type="xsd:float">0</onPeakDemandTotal>
          <onPeakConsumptionTotal xsi:type="xsd:float">0</onPeakConsumptionTotal>
          <shoulderDemandTotal xsi:type="xsd:float">0</shoulderDemandTotal>
          <shoulderConsumptionTotal xsi:type="xsd:float">0</shoulderConsumptionTotal>
          <offPeakDemandTotal xsi:type="xsd:float">0</offPeakDemandTotal>
          <offPeakConsumptionTotal xsi:type="xsd:float">0</offPeakConsumptionTotal>
          <annualDemandTotal xsi:type="xsd:float">0</annualDemandTotal>
          <annualConsumptionTotal xsi:type="xsd:float">232.33067</annualConsumptionTotal>
          <totalCharges xsi:type="xsd:float">232.33067</totalCharges>
          <month xsi:type="xsd:int">1</month>
        </item>

        <item>
          <tariff_id xsi:type="xsd:int">299</tariff_id>
          <fixedTotal xsi:type="xsd:float">0</fixedTotal>
          <consumptionTotal xsi:type="xsd:float">232.33067</consumptionTotal>
          <demandTotal xsi:type="xsd:float">0</demandTotal>
          <onPeakDemandTotal xsi:type="xsd:float">0</onPeakDemandTotal>
          <onPeakConsumptionTotal xsi:type="xsd:float">0</onPeakConsumptionTotal>
          <shoulderDemandTotal xsi:type="xsd:float">0</shoulderDemandTotal>
          <shoulderConsumptionTotal xsi:type="xsd:float">0</shoulderConsumptionTotal>
          <offPeakDemandTotal xsi:type="xsd:float">0</offPeakDemandTotal>
          <offPeakConsumptionTotal xsi:type="xsd:float">0</offPeakConsumptionTotal>
          <annualDemandTotal xsi:type="xsd:float">0</annualDemandTotal>
          <annualConsumptionTotal xsi:type="xsd:float">232.33067</annualConsumptionTotal>
          <totalCharges xsi:type="xsd:float">232.33067</totalCharges>
          <month xsi:type="xsd:int">2</month>
        </item>

        ... For each month ...

        <item>
          <tariff_id xsi:type="xsd:int">299</tariff_id>
          <fixedTotal xsi:type="xsd:float">0</fixedTotal>
          <consumptionTotal xsi:type="xsd:float">232.33067</consumptionTotal>
          <demandTotal xsi:type="xsd:float">0</demandTotal>
          <onPeakDemandTotal xsi:type="xsd:float">0</onPeakDemandTotal>
          <onPeakConsumptionTotal xsi:type="xsd:float">0</onPeakConsumptionTotal>
          <shoulderDemandTotal xsi:type="xsd:float">0</shoulderDemandTotal>
        
```

```

<shoulderConsumptionTotal xsi:type="xsd:float">0</shoulderConsumptionTotal>
<offPeakDemandTotal xsi:type="xsd:float">0</offPeakDemandTotal>
<offPeakConsumptionTotal xsi:type="xsd:float">0</offPeakConsumptionTotal>
<annualDemandTotal xsi:type="xsd:float">0</annualDemandTotal>
<annualConsumptionTotal xsi:type="xsd:float">232.33067</annualConsumptionTotal>
<totalCharges xsi:type="xsd:float">232.33067</totalCharges>
<month xsi:type="xsd:int">11</month>
</item>

<item>
<tariff_id xsi:type="xsd:int">299</tariff_id>
<fixedTotal xsi:type="xsd:float">0</fixedTotal>
<consumptionTotal xsi:type="xsd:float">232.33067</consumptionTotal>
<demandTotal xsi:type="xsd:float">0</demandTotal>
<onPeakDemandTotal xsi:type="xsd:float">0</onPeakDemandTotal>
<onPeakConsumptionTotal xsi:type="xsd:float">0</onPeakConsumptionTotal>
<shoulderDemandTotal xsi:type="xsd:float">0</shoulderDemandTotal>
<shoulderConsumptionTotal xsi:type="xsd:float">0</shoulderConsumptionTotal>
<offPeakDemandTotal xsi:type="xsd:float">0</offPeakDemandTotal>
<offPeakConsumptionTotal xsi:type="xsd:float">0</offPeakConsumptionTotal>
<annualDemandTotal xsi:type="xsd:float">0</annualDemandTotal>
<annualConsumptionTotal xsi:type="xsd:float">232.33067</annualConsumptionTotal>
<totalCharges xsi:type="xsd:float">232.33067</totalCharges>
<month xsi:type="xsd:int">12</month>
</item>
</return>
</ns5:doGetYearlyBillResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

## doGetTOU

### *Request*

```

<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" 
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:ns4="urn:onTAP"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

<SOAP-ENV:Body>
<ns4:doGetTOU>
<tariff_id xsi:type="xsd:int">430</tariff_id></ns4:doGetTOU>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

### *Responxe*

```

<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" 
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:ns4="urn:onTAP"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

<SOAP-ENV:Body>
<ns4:doGetTOUResponse>
<return>
<TOUMonths xsi:type="SOAP-ENC:Array" SOAP-ENC:arrayType="Struct[12]" SOAP-
  ENC:offset="[0]">
<item>
```

```

<MonthName xsi:type="xsd:string">January</MonthName>
<Season xsi:type="xsd:string">Winter</Season>
<PeakDays xsi:type="SOAP-ENC:Array" SOAP-ENC:arrayType="xsd:string[5]" SOAP-ENC:offset="[0]">
<item xsi:type="xsd:string">Monday</item>
<item xsi:type="xsd:string">Tuesday</item>
<item xsi:type="xsd:string">Wednesday</item>
<item xsi:type="xsd:string">Thursday</item>
<item xsi:type="xsd:string">Friday</item>
</PeakDays>
<HourEnding_01 xsi:type="xsd:string">offPeak</HourEnding_01>
<HourEnding_02 xsi:type="xsd:string">offPeak</HourEnding_02>
<HourEnding_03 xsi:type="xsd:string">offPeak</HourEnding_03>
<HourEnding_04 xsi:type="xsd:string">offPeak</HourEnding_04>
<HourEnding_05 xsi:type="xsd:string">offPeak</HourEnding_05>
<HourEnding_06 xsi:type="xsd:string">offPeak</HourEnding_06>
<HourEnding_07 xsi:type="xsd:string">offPeak</HourEnding_07>
<HourEnding_08 xsi:type="xsd:string">shoulder</HourEnding_08>
<HourEnding_09 xsi:type="xsd:string">shoulder</HourEnding_09>
<HourEnding_10 xsi:type="xsd:string">shoulder</HourEnding_10>
<HourEnding_11 xsi:type="xsd:string">shoulder</HourEnding_11>
<HourEnding_12 xsi:type="xsd:string">shoulder</HourEnding_12>
<HourEnding_13 xsi:type="xsd:string">shoulder</HourEnding_13>
<HourEnding_14 xsi:type="xsd:string">shoulder</HourEnding_14>
<HourEnding_15 xsi:type="xsd:string">shoulder</HourEnding_15>
<HourEnding_16 xsi:type="xsd:string">shoulder</HourEnding_16>
<HourEnding_17 xsi:type="xsd:string">shoulder</HourEnding_17>
<HourEnding_18 xsi:type="xsd:string">shoulder</HourEnding_18>
<HourEnding_19 xsi:type="xsd:string">shoulder</HourEnding_19>
<HourEnding_20 xsi:type="xsd:string">shoulder</HourEnding_20>
<HourEnding_21 xsi:type="xsd:string">offPeak</HourEnding_21>
<HourEnding_22 xsi:type="xsd:string">offPeak</HourEnding_22>
<HourEnding_23 xsi:type="xsd:string">offPeak</HourEnding_23>
<HourEnding_24 xsi:type="xsd:string">offPeak</HourEnding_24>
</item>

<item>
<MonthName xsi:type="xsd:string">February</MonthName>
<Season xsi:type="xsd:string">Winter</Season>
<PeakDays xsi:type="SOAP-ENC:Array" SOAP-ENC:arrayType="xsd:string[5]" SOAP-ENC:offset="[0]">
<item xsi:type="xsd:string">Monday</item>
<item xsi:type="xsd:string">Tuesday</item>
<item xsi:type="xsd:string">Wednesday</item>
<item xsi:type="xsd:string">Thursday</item>
<item xsi:type="xsd:string">Friday</item>
</PeakDays>
<HourEnding_01 xsi:type="xsd:string">offPeak</HourEnding_01>
<HourEnding_02 xsi:type="xsd:string">offPeak</HourEnding_02>
<HourEnding_03 xsi:type="xsd:string">offPeak</HourEnding_03>
<HourEnding_04 xsi:type="xsd:string">offPeak</HourEnding_04>
<HourEnding_05 xsi:type="xsd:string">offPeak</HourEnding_05>
<HourEnding_06 xsi:type="xsd:string">offPeak</HourEnding_06>
<HourEnding_07 xsi:type="xsd:string">offPeak</HourEnding_07>
<HourEnding_08 xsi:type="xsd:string">shoulder</HourEnding_08>
<HourEnding_09 xsi:type="xsd:string">shoulder</HourEnding_09>
<HourEnding_10 xsi:type="xsd:string">shoulder</HourEnding_10>
<HourEnding_11 xsi:type="xsd:string">shoulder</HourEnding_11>
<HourEnding_12 xsi:type="xsd:string">shoulder</HourEnding_12>
<HourEnding_13 xsi:type="xsd:string">shoulder</HourEnding_13>
<HourEnding_14 xsi:type="xsd:string">shoulder</HourEnding_14>
<HourEnding_15 xsi:type="xsd:string">shoulder</HourEnding_15>
<HourEnding_16 xsi:type="xsd:string">shoulder</HourEnding_16>

```

```

<HourEnding_17 xsi:type="xsd:string">shoulder</HourEnding_17>
<HourEnding_18 xsi:type="xsd:string">shoulder</HourEnding_18>
<HourEnding_19 xsi:type="xsd:string">shoulder</HourEnding_19>
<HourEnding_20 xsi:type="xsd:string">shoulder</HourEnding_20>
<HourEnding_21 xsi:type="xsd:string">offPeak</HourEnding_21>
<HourEnding_22 xsi:type="xsd:string">offPeak</HourEnding_22>
<HourEnding_23 xsi:type="xsd:string">offPeak</HourEnding_23>
<HourEnding_24 xsi:type="xsd:string">offPeak</HourEnding_24>
</item>

... For each month ...

<item>
<MonthName xsi:type="xsd:string">November</MonthName>
<Season xsi:type="xsd:string">Winter</Season>
<PeakDays xsi:type="SOAP-ENC:Array" SOAP-ENC:arrayType="xsd:string[5]" SOAP-ENC:offset="[0]">
<item xsi:type="xsd:string">Monday</item>
<item xsi:type="xsd:string">Tuesday</item>
<item xsi:type="xsd:string">Wednesday</item>
<item xsi:type="xsd:string">Thursday</item>
<item xsi:type="xsd:string">Friday</item>
</PeakDays>
<HourEnding_01 xsi:type="xsd:string">offPeak</HourEnding_01>
<HourEnding_02 xsi:type="xsd:string">offPeak</HourEnding_02>
<HourEnding_03 xsi:type="xsd:string">offPeak</HourEnding_03>
<HourEnding_04 xsi:type="xsd:string">offPeak</HourEnding_04>
<HourEnding_05 xsi:type="xsd:string">offPeak</HourEnding_05>
<HourEnding_06 xsi:type="xsd:string">offPeak</HourEnding_06>
<HourEnding_07 xsi:type="xsd:string">offPeak</HourEnding_07>
<HourEnding_08 xsi:type="xsd:string">shoulder</HourEnding_08>
<HourEnding_09 xsi:type="xsd:string">shoulder</HourEnding_09>
<HourEnding_10 xsi:type="xsd:string">shoulder</HourEnding_10>
<HourEnding_11 xsi:type="xsd:string">shoulder</HourEnding_11>
<HourEnding_12 xsi:type="xsd:string">shoulder</HourEnding_12>
<HourEnding_13 xsi:type="xsd:string">shoulder</HourEnding_13>
<HourEnding_14 xsi:type="xsd:string">shoulder</HourEnding_14>
<HourEnding_15 xsi:type="xsd:string">shoulder</HourEnding_15>
<HourEnding_16 xsi:type="xsd:string">shoulder</HourEnding_16>
<HourEnding_17 xsi:type="xsd:string">shoulder</HourEnding_17>
<HourEnding_18 xsi:type="xsd:string">shoulder</HourEnding_18>
<HourEnding_19 xsi:type="xsd:string">shoulder</HourEnding_19>
<HourEnding_20 xsi:type="xsd:string">shoulder</HourEnding_20>
<HourEnding_21 xsi:type="xsd:string">offPeak</HourEnding_21>
<HourEnding_22 xsi:type="xsd:string">offPeak</HourEnding_22>
<HourEnding_23 xsi:type="xsd:string">offPeak</HourEnding_23>
<HourEnding_24 xsi:type="xsd:string">offPeak</HourEnding_24>
</item>

<item>
<MonthName xsi:type="xsd:string">December</MonthName>
<Season xsi:type="xsd:string">Winter</Season>
<PeakDays xsi:type="SOAP-ENC:Array" SOAP-ENC:arrayType="xsd:string[5]" SOAP-ENC:offset="[0]">
<item xsi:type="xsd:string">Monday</item>
<item xsi:type="xsd:string">Tuesday</item>
<item xsi:type="xsd:string">Wednesday</item>
<item xsi:type="xsd:string">Thursday</item>
<item xsi:type="xsd:string">Friday</item></PeakDays>
<HourEnding_01 xsi:type="xsd:string">offPeak</HourEnding_01>
<HourEnding_02 xsi:type="xsd:string">offPeak</HourEnding_02>
<HourEnding_03 xsi:type="xsd:string">offPeak</HourEnding_03>
<HourEnding_04 xsi:type="xsd:string">offPeak</HourEnding_04>

```

```
<HourEnding_05 xsi:type="xsd:string">offPeak</HourEnding_05>
<HourEnding_06 xsi:type="xsd:string">offPeak</HourEnding_06>
<HourEnding_07 xsi:type="xsd:string">offPeak</HourEnding_07>
<HourEnding_08 xsi:type="xsd:string">shoulder</HourEnding_08>
<HourEnding_09 xsi:type="xsd:string">shoulder</HourEnding_09>
<HourEnding_10 xsi:type="xsd:string">shoulder</HourEnding_10>
<HourEnding_11 xsi:type="xsd:string">shoulder</HourEnding_11>
<HourEnding_12 xsi:type="xsd:string">shoulder</HourEnding_12>
<HourEnding_13 xsi:type="xsd:string">shoulder</HourEnding_13>
<HourEnding_14 xsi:type="xsd:string">shoulder</HourEnding_14>
<HourEnding_15 xsi:type="xsd:string">shoulder</HourEnding_15>
<HourEnding_16 xsi:type="xsd:string">shoulder</HourEnding_16>
<HourEnding_17 xsi:type="xsd:string">shoulder</HourEnding_17>
<HourEnding_18 xsi:type="xsd:string">shoulder</HourEnding_18>
<HourEnding_19 xsi:type="xsd:string">shoulder</HourEnding_19>
<HourEnding_20 xsi:type="xsd:string">shoulder</HourEnding_20>
<HourEnding_21 xsi:type="xsd:string">offPeak</HourEnding_21>
<HourEnding_22 xsi:type="xsd:string">offPeak</HourEnding_22>
<HourEnding_23 xsi:type="xsd:string">offPeak</HourEnding_23>
<HourEnding_24 xsi:type="xsd:string">offPeak</HourEnding_24>
</item>
</TOUMonths>
</return>
</ns4:doGetTOUResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```